

Cómo hacer una página web con HTML



por **Berto López**, últ. act. el 7 noviembre, 2019

Categorías: [HTML & CSS](#)

En el post de hoy voy a enseñarte **cómo hacer una página web con HTML**, pero antes de meterme en harina, **es necesario aclarar un concepto que es vital** en este contexto que es precisamente el de **“página web”** versus **«sitio web»**.

La razón de ello es que se suele utilizar con frecuencia el término **“página web”** cuando en realidad se está hablando de un **“sitio web”** y esto puede dar lugar a confusión porque las dos cosas **no son lo mismo**.

No te preocupes porque en este post vas a aprender las dos, pero debes tener claro que **son dos cosas diferentes**.

Tabla de contenidos [ocultar]

- Conceptos básicos de páginas web
 - Diferencias entre una página web y un sitio web
 - Qué es el HTML (HyperText Markup Language)
 - Conceptos HTML fundamentales: Etiquetas, elementos y atributos HTML
 - ¿Es una buena idea hacer un sitio web entero con HTML?
- ¿Qué cosas necesitas dominar para hacer páginas web?
 - Etiquetas básicas de estructura y cabecera
 - Etiquetas para títulos, párrafos, enlaces, imágenes y formatos básicos de texto
 - Etiquetas para dar más estructura: divisiones, tablas y listas
 - Interaccionar con el usuario con formularios
- Crear una página web con HTML
 - Ejemplo práctico de una página web sencilla
 - Cómo analizar otras páginas web para tomar ideas

- Cómo hacer páginas web en HTML online
- Cómo hacer páginas web en HTML offline
- Cómo crear un sitio web entero con páginas HTML
 - La portada (home page) de tu sitio web
 - Cómo organizar las diferentes zonas de tu web
 - Cómo organizar los recursos de tu web
 - Ejemplo de estructura genérica para una web
- Cómo preparar el HTML de tus páginas web para Google
 - Nunca uses más de una etiqueta H1
 - Otros elementos claves de SEO On Page
- Concluyendo

Conceptos básicos de páginas web

¿Cuál es entonces la diferencia entre una página web y un sitio web?

Diferencias entre una página web y un sitio web

Una página web es un **único documento electrónico** que se puede visualizar en un navegador y que está codificado en lenguaje HTML. Cada página web tiene una dirección web (una “URL”) única.

Por ejemplo:

<https://www.hostingatope.com/mejores-herramientas-para-webmasters-internet/>

En cambio, un sitio web es el **conjunto de páginas web** pertenecientes a un **mismo dominio**.

Por ejemplo, en el caso de este blog:

Hosting a Tope es el **sitio web** y éste a su vez se compone **todas las páginas web** que se ubican en el dominio [hostingatope.com](https://www.hostingatope.com), incluida la del ejemplo anterior. Cada una de ellas tendrá una URL diferenciada que pertenece al dominio [hostingatope.com](https://www.hostingatope.com).

Por otra parte, lo que ves en el dominio hostingatope.com es a su vez una página, la que corresponde precisamente a la URL que en este caso es el dominio sin más. Es la **página principal** del sitio, también conocida como **portada** o **home page**, según la terminología que prefiera usar cada uno.

Para que tu página web se vea finalmente en Internet, necesitas **un dominio y un servidor conectado a Internet**. Esto lo puedes conseguir con servicios gratuitos como, por ejemplo, Blogger o WordPress.com, todos ellos, muy limitados o contratando un **hosting**

Qué es el HTML (HyperText Markup Language)

HTML son las siglas de “*HyperText Markup Language*”, lo que se puede traducir en español a “*Lenguaje de marcado de hipertexto (texto con enlaces)*”.

Eso quiere decir que permite “marcar” el texto mediante unas **etiquetas especiales** que le añaden semántica.

Por ejemplo:

```
<h1>Cómo hacer una página web con HTML</h1>
```

¿Qué ha pasado aquí? Pues que simplemente añadiendo la etiqueta **<h1>** le hemos añadido una **información semántica** muy útil a la frase que indica que esto no es una frase cualquiera sino un **título** (fíjate en la “h” de “header” en inglés) de **nivel 1**.

Esto es muy útil porque, por ejemplo, le sirve al navegador web para aplicar ciertas **pautas de diseño** que faciliten la lectura como diferentes tamaños de letra a los títulos según su nivel jerárquico, de modo que destaquen en la lectura.

Otro ejemplo es el buscador de Google. Si alguien busca, por ejemplo, “*cómo hacer una página web*”, al ver que son palabras que coinciden con el título principal de la página, esto le indicará a Google que esa página tiene que ver probablemente mucho con lo que el usuario está buscando.

Como ves con estos dos simples ejemplos, el marcado que proporciona HTML permite hacer muchas más cosas con estos contenidos que con un texto normal y corriente sin marcado. Ese es el propósito básico del HTML, aparte de permitir enlazar las páginas entre sí y crear así la “web” de Internet.

Conceptos HTML fundamentales: Etiquetas, elementos y atributos HTML

A lo largo del tiempo se han desarrollado diferentes versiones de HTML siendo la más usada la versión **HTML 4.01**. Poco a poco está siendo relevada por la versión HTML 5 que introduce nuevas etiquetas que amplían las posibilidades que tenía HTML 4.01.

Pero en este punto, lo mejor es que ya empecemos a meternos en harina en ver el código HTML de una página web.

Por ejemplo: una versión extremadamente simplificada de este post, tendría un código HTML como el siguiente:

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Cómo hacer una página web con HTML</title>
  </head>
  <body>
    <h1>Cómo hacer una página web con HTML</h1>
    <p> En el post de hoy voy a enseñarte <strong>cómo
hacer una página web con HTML</strong>, pero antes ...</p>
    <h2>Conceptos básicos sobre páginas web</h2>
    <p>¿Cuál es entonces la diferencia entre una página
web y un sitio web?...</p>
    <h3>Diferencias entre una página web y un sitio
web</h3>
    <p>Una <a
href="https://es.wikipedia.org/wiki/P%C3%Algina_web">págin
a web</a> es un <strong>único documento
electrónico</strong> que...</p>
  </body>
</html>
```

El ejemplo empieza con una etiqueta especial que es **<!DOCTYPE HTML>**, es simplemente una etiqueta especial que tiene que ir al principio de todo fichero HTML e identifica al fichero como un documento en formato HTML.

Lo que sigue son etiquetas anidadas que forman una estructura jerárquica (excepto la línea del DOCTYPE). Fíjate que siempre hay una **etiqueta de**

apertura y una **etiqueta de cierre** que es la misma etiqueta, pero con una barra invertida delante del nombre de la etiqueta.

Estas etiquetas, junto con el contenido que envuelven, forman lo que se conoce como un elemento HTML.

En el ejemplo podrás observar que hay una estructura básica compuesta por las etiquetas **<html>**, **<head>** y **<body>**. Esto es la estructura básica de cualquier página web.

<html> es el elemento raíz de la página, es simplemente el elemento del que tienen que “colgar” todos los demás, no tiene más función que esa. **<head>** es la cabecera de la página, una sección con **información descriptiva** de la página, etc. que no se visualiza como tal en la página, eso le corresponde a la etiqueta **<body>**. En **<body>** irán **los elementos que vemos realmente dentro del navegador** como tales.

Fíjate también en cómo se abren y cierran las etiquetas, por ejemplo, la etiqueta “**<title>**” (etiqueta de apertura) y “**</title>**” (etiqueta de cierre). A su vez, estas etiquetas pueden contener a otras etiquetas, como es el caso de **<head>** que es la que encierra a la etiqueta o elemento “**title**”.

Si seguimos, verás que varios niveles de títulos con etiquetas **<h1>** hasta **<h3>**. Esto sería lo equivalente, por ejemplo, a los estilos de títulos “Título 1”, “Título 2” y “Título 3” en Microsoft Word. En HTML dispones de un total de 6 niveles, de h1 a h6.

Por otra parte, puedes ver varias etiquetas **<p>**, éstas corresponden a los párrafos que vienen a continuación de los títulos y para las cuales hemos puesto solamente un pequeño extracto de texto como ejemplo.

Fíjate especialmente en las etiquetas tipo **<a>** que son los enlaces. Ésta es, si cabe, la etiqueta por excelencia del HTML, la que permite crear los hiperenlaces que son lo que hace de la web la web.

Con esto puedo introducir, además, otro concepto fundamental del HTML: **los atributos**.

Si echas un vistazo con atención, en el caso de la etiqueta **<a>**, antes del corchete de cierre vienen dos palabras: “**href**” y “**target**”. Estas palabras son dos **atributos** y sirven para especificar **parámetros adicionales** en una etiqueta. De esta manera le podemos dar mucho más juego a las etiquetas.

En ejemplo concreto el atributo “href” indica la URL de la página enlazada y con el atributo “target” indicamos que, además, queremos que cuando el usuario haga clic en él, la página correspondiente se abra en otra pestaña del navegador.

¿Es una buena idea hacer un sitio web entero con HTML?

Ahora que ya te puedes hacer una idea concreta de lo que es una página HTML, quiero mencionar también otra cuestión que me parece **muy importante** plantearte antes de que decidas a ponerte a hacer páginas web con HTML:

Aunque todas las páginas web estén compuestas por HTML, **hoy en día ya no es necesario picar «a pelo» el HTML que has visto antes**. Es más, suele ser generalmente la peor opción, la más laboriosa y la menos productiva.

Hoy en día, si quieres crear un sitio web, la fórmula normal suele ser algún tipo de herramienta como **WordPress, Blogger, Wix**, etc. que te permiten crear los contenidos en HTML de una manera **mucho más eficiente**.

Por una parte, para crear los contenidos, usando **editores visuales** parecidos a los de herramientas como Microsoft Word, por ejemplo.

Por otra parte, para lo que es el diseño en conjunto de la web, dispones de las «plantillas» de WordPress que son, en definitiva, plantillas de diseño personalizables:

Eso sí, para explotar todas las posibilidades que esto ofrece, **vas a necesitas WordPress con hosting propio**, la opción gratuita de WordPress está **muy limitada**, al igual que las versiones gratuitas de las demás plataformas alternativas.

Además, con este tipo de herramientas puedes añadir mucha funcionalidad (comentarios, feed RSS, etc.) que con una página HTML en principio no puedes implementar o que, al menos, supondrían un esfuerzo muy grande y necesitar programar con lenguajes de programación avanzados como JavaScript.

¿Quiere decir que eso no tiene sentido que aprendas HTML?

En absoluto.

La razón es muy simple: aunque, en general, trabajes con herramientas como las mencionadas siempre hay ocasiones en las cuales para una cosa concreta que quieras hacer te será más cómodo hacer eso en concreto con HTML. Por otra parte, es muy bueno que sepas HTML para que puedas revisar el HTML que generan ya que algunas veces contiene fallos o artefactos que no convienen y conviene “limpiarlo” un poco si quieres dejar el contenido perfecto.

Además, tengo una **buena noticia: aprender HTML es bastante más fácil de lo que piensas.**

Así que sí o sí te animo a que aprendas HTML y CSS porque te cambia la vida en cuanto a lo que serás capaz de hacer en un sitio web.

¿Qué cosas necesitas dominar para hacer páginas web?

La cantidad de elementos y atributos en HTML es enorme. Es un lenguaje muy potente y sólo tienes que ver la cantidad de páginas web espectaculares que existen.

Por eso, aquí voy a hacer **una selección** de los que serían a mi juicio los **elementos más importantes**. Con esto tienes más que suficiente para crear incluso páginas web avanzadas y ya tienen que ser necesidades muy específicas para que realmente necesites más.

Como referencia con todas las etiquetas HTML, te dejo [en esta referencia](#) en la que encontrarás un listado completo, junto con los detalles de cada una.

Y si quieres aprender HTML a fondo, te recomiendo este libro online gratuito:

Introducción a XHTML

Etiquetas básicas de estructura y cabecera

Empezamos por los elementos que conforman la estructura básica que vimos antes (**<html>**, **<head>** y **<body>**) y elementos que te resultarán ahora un

tanto “abstractos” con información adicional sobre la página (<meta>, <title> y <link>) y que irán dentro de la etiqueta <head>.

Salvo de <title> (que determina el nombre de la página que verás en la pestaña del navegador), de estos últimos, por ahora no te preocupes demasiado. Son un poco más avanzados e iremos hablando más de ellos en los momentos adecuados.

- Elemento raíz: <html>
- Información del documento: <head>
- Cuerpo del documento: <body>
- Meta-información: <meta>
- Título de página: <title>
- Enlace a recursos (hojas de estilo, etc.): <link>

Etiquetas para títulos, párrafos, enlaces, imágenes y formatos básicos de texto

Éstas son las etiquetas para crear la estructura básica de una página HTML:

- Títulos: <h1> – <h6>
- Párrafo: <p>
- Hiperenlace: <a>
- Imagen:
- Sección de texto:
- Énfasis: ,
- Salto de línea:

Etiquetas para dar más estructura: divisiones, tablas y listas

Para darle una estructura un poco más sofisticada e interesante a una página, las listas y tablas son elementos totalmente esenciales.

Por otra parte, las etiquetas <div> permiten crear divisiones o zonas en una página. Una división básica de una página podría ser, por ejemplo, una cabecera, el cuerpo del contenido, una barra lateral y un pie. Esto es, por ejemplo, el formato básico de un blog típico.

Aunque para sacarles realmente partido necesitas aprender también CSS ya que es con las reglas CSS cómo se “maqueta” esta parte, es decir, se definen cosas como la alineación de una división, sus dimensiones, etc.

En HTML dispones de las siguientes etiquetas para crearlas:

- **Divisiones:** <div>
- **Tablas:** <table>, <tr> y <td>
- **Listas:** ,
- **Ítems de lista:**

Interaccionar con el usuario con formularios

HTML también está preparado para crear interacciones sencillas con los usuarios mediante formularios. Eso sí, si usas formularios en el lado del servidor necesitas algún tipo de programa que reciba esa información y la sepa procesar para guardar, por ejemplo, la información que recojas en una base de datos.

Esto es, por cierto, otra ventaja de usar aplicaciones como WordPress para tu web, aquí estas cosas ya las tienes hechas y puedes crear formularios para recoger sus datos en una base de datos sin saber nada de programación.

Ejemplos típicos y muy simples serían un formulario de contacto a través del cual se genera un email que recibes en tu cuenta de email o un formulario de pedidos para que le gente te envíe sus pedidos.

Para los formularios se usan las etiquetas **<form>** e **<input>**, pero son un poco más complicadas de usar y, por tanto, cómo usarlas exactamente se sale un poco del marco de este post. Te remito a este enlace para este tema.

- **Formulario:** <form>
- **Campo:** <input>

Crear una página web con HTML

Vistos los elementos más importantes, ya estamos en condiciones para crear una página web real sencilla.

Crear una página web con HTML consiste básicamente en crear un fichero y editar un contenido que será codificado en el lenguaje HTML. Para ello existen diversas herramientas.

Ejemplo práctico de una página web sencilla

Para ello te voy a poner **un ejemplo concreto** que tendrá como resultado lo que ves en la captura de imagen siguiente.

En el ejemplo, como puedes ver, **me he inspirado en los blogs** y he creado una página HTML sencilla con una organización y estética muy similar a lo que puedes ver en muchos blogs. Cuenta con una cabecera, un cuerpo que se divide en el contenido principal (los posts), una barra lateral con un anuncio y un pie.



Captura de pantalla del resultado de nuestro ejercicio práctico. Como se puede ver, a primera vista parece el típico post de un blog.

A continuación, te dejo también un **extracto del código HTML** de esta página para que te puedas hacer ya una idea de cómo está hecha.

También es necesario comentar que para hacer una página un poco más real y no excesivamente simplista, he utilizado también hojas de estilo CSS que se encargan básicamente de la estética (tipos de letra, tamaños, colores, etc.) y de la maquetación de la página.

En este post no nos vamos a meter con la parte CSS, nos centraremos en la parte HTML

Ahora me gustaría que te fijases en particular que para crear esta página nos hemos apañado perfectamente con las etiquetas que hemos repasado un poco más arriba.

```
<!DOCTYPE html>
<html>
<head lang="en">
    <meta charset="UTF-8">
    <title>Ejemplo de una página web de nivel medio-
basico</title>
    <link rel="stylesheet" type="text/css"
href="style.css">
</head>
<body>
    <div class="cabecera">
        <a href="/"></a>
    </div>
    <div class="cuerpo">
        <div class="contenido">
            <h1>3 Consejos para hacer páginas web
bien</h1>
            <h2>1. No te limites a herramientas
visuales</h2>
            <p>Existen <a target="_blank"
href="http://www.quackit.com/html/online-html-
editor/">muchas herramientas</a>, incluso gratuitas, que
te permiten crear páginas web sin saber realmente
HTML.</p>
            <p>...</p>
```

```

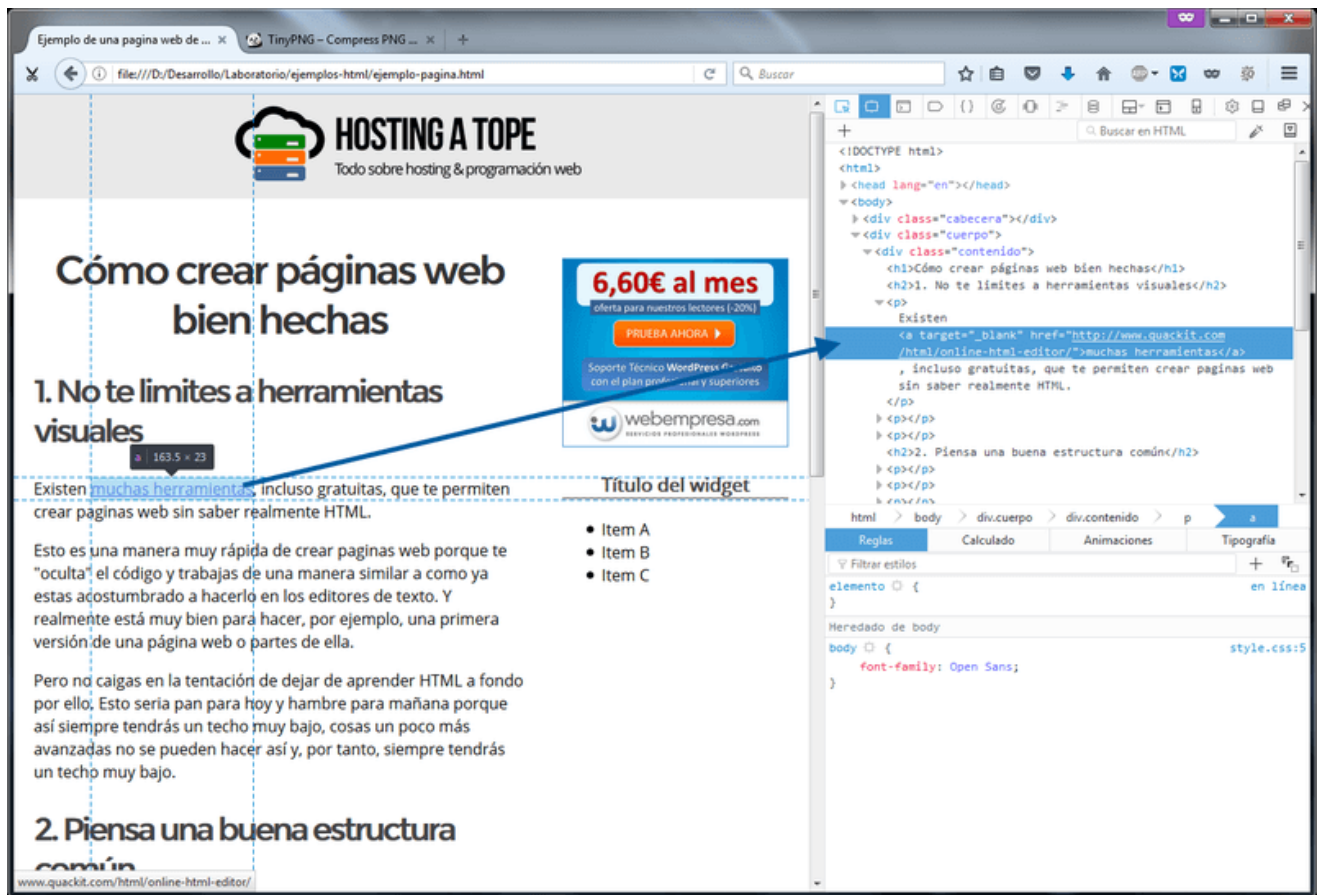
        <h2>2. Piensa una buena estructura común</h2>
        <p>Antes de crear una página web, procura
pensar una estructura lógica común compuesta por etiquetas
de divisiones.</p>
        <p>...</p>
    </div>
    <div class="barra-lateral">
        <a target="_blank" href="/go/webempresa"></a>
    </div>
</div>
<div class="pie">
    <p>© Copyright Hosting a Tope 2017</p>
</div>
</body>
</html>

```

Cómo analizar otras páginas web para tomar ideas

Ahora que ya tienes las bases de cómo crear páginas web con HTML, te animo a que mires cómo están hechas otras páginas. Y para ello hay una **herramienta maravillosa**: los **inspectores de código de los principales navegadores**.

Son todos muy parecidos y los encontrarás en **Firefox**, **Chrome** o el **Internet Explorer** (y probablemente en alguno más entre los menos conocidos). En todos ellos se activan con la tecla F12. La principal ventaja que te aportarán es que en vez de visualizar el código HTML simplemente como texto, lo visualizan como árbol, colorean las palabras clave (etiquetas, etc.) y te permiten seleccionar elementos determinados desde la página.



Captura de pantalla en la que puede ver el inspector de código de Firefox en acción. Con el selector de elementos hemos seleccionado un elemento en la página (un enlace) y el inspector nos muestra en la parte derecha el código HTML correspondiente. Una maravilla para «destripar» páginas ajenas 😊

Es decir, si quieres saber cómo está hecho un determinado elemento de una página web que estás viendo, no tienes más que seleccionar ese elemento y la herramienta te mostrará en el árbol del código HTML ese elemento en concreto. Es fantástico para aprender y tomar ideas.

Pero una advertencia:

Cuando veas ejemplos reales probablemente te asustes, en día las páginas web de cualquier blog suelen ser muy grandes y complejas, muchísimo más que el ejemplo que he puesto aquí, aunque a simple vista de pueda parecer que son similares.

No voy a entrar ahora en las razones de por qué esto es así porque no son importantes para nosotros, sino que simplemente te quiero lanzar un mensaje de tranquilidad: No pasa nada, es simplemente un poco más incómodo que

analizar la página de nuestro ejemplo, pero si las analizas con calma, reconocerás perfectamente los elementos que hemos ido comentando aquí y que son los que realmente importan.

Ahora bien, si puedes encontrar y analizar páginas sencillas al principio, mejor que mejor.

Cómo hacer páginas web en HTML online

Por suerte, hoy en día encuentras herramientas online que no requieren la instalación de un programa en tu PC y que funcionan en cualquier entorno ya sea un PC con Windows, un ordenador Mac o un ordenador con Linux.

Lo que sí te recomiendo es que no intentes utilizar tablets o dispositivos similares para ello, aunque funciona en la mayoría de los casos, con dispositivos muy incómodos para trabajar con estas cosas.

Aquí te dejo un listado de varias herramientas online con diferentes enfoques y fortalezas específicas, además, todas son gratuitas.

1. **El editor online de Quackit**: Ésta es mi herramienta preferida, la que uso habitualmente cuando tengo necesidad de crear algo de HTML. Es un editor sencillo que te permite crear HTML de manera visual o con código. Lo mejor es que el código HTML que genera, a diferencia de muchas otras, es muy “limpio”. Además, puedes cambiar en todo momento entre el modo de edición visual y en texto. Ya me gustaría que, por ejemplo, el editor por defecto de WordPress funcionase así de bien.
2. **HTML Instant**: Éste es muy simple comparado con el anterior, pero tiene una ventaja que me parece muy interesante para cuando estás aprendiendo: Tiene una división de pantalla en dos mitades (verticales) donde en una ves el código HTML y en la otra el resultado visual. Super-práctico para practicar con tus primeros pasos.
3. **Best Online WYSIWYG HTML Editor**: Éste es el editor más sofisticado y potente de todos, pero tanto que no te lo recomendaría como herramienta para empezar, sino para más adelante J

Cómo hacer páginas web en HTML offline

Si prefieres utilizar herramientas gratuitas offline para editar tus páginas HTML, aquí te dejo también dos opciones, una básica y otra más avanzada:

1. **Notepad++**: Éste es un editor sencillo y cómodo para crear ficheros HTML, soporta lo básico como una sintaxis coloreada para facilitar la lectura del código y está muy bien para hacer tus primeros pinitos con páginas muy sencillitas. La desventaja es que sólo puedes trabajar a nivel de código, es decir, no puedes visualizar la página como tal, esto lo tienes que hacer en un navegador (lo cual tampoco es que sea un gran problema, en realidad...).
2. **Brackets**: Brackets es un proyecto open source (código abierto) muy original de Adobe. Es un editor de HTML, CSS y JavaScript creado con HTML, CSS y JavaScript. Es una herramienta muy potente que lógicamente ya requiere bastante más aprendizaje que Notepad++, pero también es otra dimensión de herramienta. Recomendaba para cuando empieces a “controlar” un poco.

Aquí te dejo un vídeo de introducción sobre Brackets:

Cómo crear un sitio web entero con páginas HTML

Si te dispones a crear un sitio web con páginas HTML hay que hablar también un poco de organización, de lo contrario, la cosa se te puede ir rápidamente de las manos.

La portada (home page) de tu sitio web

Todo sitio web tiene su página principal, la que coincide con el dominio del sitio. Conviene que la convención suele ser que la página web que se visualice en el dominio se llame index.html.

Se pueden configurar otras opciones, pero por defecto cualquier servidor web te debería reconocer index.html sin problemas y no te complicas la vida.

Cómo organizar las diferentes zonas de tu web

Si tienes una cantidad considerable de páginas, conviene que pienses en organizar zonas de tu web en forma de directorios que sigan una cierta lógica, por ejemplo, por temas o categorías. De lo contrario, te resultará cada vez más mantener una visión coherente de tu web.

Cómo organizar los recursos de tu web

Una página web no se compone solamente del fichero HTML principal, sino todos los recursos que integra: las hojas de estilo, imágenes y en ocasiones código en JavaScript.

Por tanto, de nuevo, es importante que organices esto bien en directorios e incluso subdirectorios si lo ves conveniente. De lo contrario, tener todo este mogollón de ficheros entremezclados en el directorio raíz hará tu web con el tiempo simple y llanamente inmanejable.

Ejemplo de estructura genérica para una web

Para concretar todo esto, te voy a proponer una estructura bastante “estándar” que vas a ver reflejada en muchas webs:

Como puedes ver, es muy simple:

- El fichero “index.html” para la página de portada (la “home page”).
- La carpeta “css” para las hojas de estilo, la carpeta “images” para las imágenes y la carpeta “js” para los ficheros de código JavaScript.
- Y, por último, las carpetas “categoría1” y “categoría2” equivaldrían a las carpetas destinadas a organizar los diferentes ficheros de las diferentes páginas.

Cómo preparar el HTML de tus páginas web para Google

Y ya por último no quiero concluir este post sin hablar también un poco de cómo posicionar tu web en Google ya que si creas páginas web es para que la gente las vea y una de las fuentes de tráfico más importantes, sino la más importantes sigue siendo el tráfico de las búsquedas en Google.

Para vamos a repasar las claves básicas del SEO On Page que se refiere a la optimización a nivel de la propia página y cómo se refleja esto a nivel HTML.

Quiero dejarte también claro que esto no pretende ser un curso de SEO, para eso echa mejor un vistazo a este post:

Aquí quiero resaltar solamente algunos aspectos muy concretos de HTML que están estrechamente relacionados con el SEO.

Nunca uses más de una etiqueta H1

La primera cuestión, y que destaca por encima de las demás, es que le dejes muy claro a Google cuáles son las palabras principales para las cuales tu página web debería aparecer en los resultados de búsquedas. Para eso te va a ser útiles, sobre todo, las etiquetas `<title>` y `<h1>`.

El “problema” es que puedes usar varias etiquetas `<h1>` si quieres, pero si quieres hacer bien tu SEO, deberías usar **solo un título `<h1>`** en cada página.

¿Por qué? Porque el `<h1>` refleja el título principal de tu página y en él deberías usar las palabras que quieres que Google identifique como palabras clave principales.

Imagínate un libro con tres títulos diferentes en su portada. ¿Sabrías adivinar cuál va a ser el tema de ese libro? No, ¿verdad? Pues eso es el equivalente a tener varias etiquetas `<h1>` en una página.

Por tanto, usa solamente una etiqueta `<h1>` en la página y te curas en salud. El contenido de esa etiqueta puede ser por cierto el mismo que el de la etiqueta `<title>` o parecido. En todo caso, conviene que las palabras clave aparezcan también en `<title>`.

Otros elementos claves de SEO On Page

Google valora también otros elementos de la página para decidir en qué lugar posicionarte, aunque no se sabe exactamente con qué peso ya que mantienen un estricto secreto sobre los detalles del funcionamiento de su algoritmo de posicionamiento.

La URL

El primer elemento a citar es la URL. Aquí simplemente te quiero comentar que conviene que repitas las palabras clave en la URL separando estas por guiones. No solamente es un formato que le gusta a Google, sino que hace la URL también más inteligible y agradable para el lector.

Además, si tienes un sitio temático, compuesto por páginas estáticas HTML, puede ser interesante que uses carpetas con palabras clave que se reflejen en la URL.

Por ejemplo:

- misitio.com/dietas/como-hacer-una-dieta-dukan.html
- misitio.com/recetas/pollo-al-curry.html

Una organización así va a favorecerte en búsquedas como “receta pollo al curry” o “como hacer una dieta dukan”.

Los subtítulos

No cuides las palabras clave solamente en los títulos h1, sino también, al menos, en los títulos h2. Tiene bastante menos peso, pero todo suma. Además, puedes aprovechar para “apuntar” a variantes de tu título principal que sabes que también son búsquedas que hace la gente en Google.

La meta-descripción

La meta-descripción es un caso especial porque es el texto que verás debajo del título en los resultados de búsqueda de Google.

Es una etiqueta que se sitúa dentro de la etiqueta <head> en la cabecera de la página.

Aquí puedes ver un ejemplo:

```
<head>
  <meta name="description" content="He usado muchos
hostings. Aquí te cuento cuál es el mejor hosting del 2015
en España la manera de conseguir el máximo descuento :) ">
</head>
```

Se debate si Google tiene realmente en cuenta las palabras clave de este texto o no, aunque desde luego no puede hacer daño que las metas.

Pero sobre lo que no hay duda es la importancia del texto en sí puesto que al aparecer en los resultados de búsqueda será, junto con el título, determinante

para que hagan clic en tu página o no. Y eso sí que se sabe que Google lo valora en gran medida para determinar en qué posición colocar cada resultado.

Por tanto, directa o indirectamente es muy importante que incluyas una meta-descripción y que la cuides si quieres hacer un buen trabajo de SEO.

El contenido

Y, por último, tendrás que usar las palabras clave en el contenido. Aquí se suele dar el consejo de buscar una densidad de, al menos, un 1%. Es decir, que de cada 100 palabras, una sea correspondiente a la palabra (o frase) clave para la cual quieres posicionar el contenido.

Concluyendo

Como ves, HTML es un lenguaje relativamente fácil de aprender, incluso para gente sin un “background” técnico.

Como todo, si te metes a fondo es un mundo, pero lo bueno que tiene es que dominando solamente una pequeña parte (lo que he expuesto aquí básicamente), junto con unos conocimientos básicos de CSS, que son imprescindibles, ya podrás empezar a hacer cosas bastante interesantes.

Ahora bien, por las razones que ya expuse antes, habiendo opciones alternativas como WordPress y otras herramientas, yo jamás haría una web “a pelo” en HTML.

No obstante, eso no quiere decir que no te merezca la pena aprender seriamente HTML, si merece la pena y mucho.

La razón es que todas estas herramientas te solucionan el 90-95% del problema, pero siempre habrá cosas (una tabla un poco complicada, un cuadro con un formato especial, un banner que quieres repetir en diferentes ubicaciones del sitio, etc.) para las cuales notas que la herramienta se te queda corta.

Aquí es dónde vas a notar una gran diferencia porque hay un mundo entre controlar tu web al 90-95% o controlarla al 100% y poder implementar cualquier cosa que se te antoje. Esa es la puerta que te abre el dominio de HTML y CSS.

